



字符串重排 (reorder)

清华大学 吴翼

题意简述

- 给定 N 个字符串 $S_1 \dots S_N$ ，将其排序
- 设排序结果为 $S_{p_1}, S_{p_2}, \dots, S_{p_n}$
- 给定 Q 个任务，第 i 个任务要求在排序结果中， S_{a_i} 恰好在 S_{b_i} 前一个
- 第 i 个任务的奖励是 2^i

题意简述

- 第一问：
- 求相邻两项最长公共前缀平方和 W 的最大值

$$W(P) = \sum_{i=2}^N LCP(S_{p_{i-1}}, S_{p_i})^2$$

题意简述

- 第二问：
- 输出使得 W 最大的同时能够获得最多奖励的排序结果

数据规模

- 10%的数据, $N \leq 10$, $Length(S_i) \leq 50$, $Q = 1$
- 20%的数据, $N \leq 50$, $Length(S_i) \leq 50$, $Q = 1$
- 50%的数据, $N \leq 1000$, $Length(S_i) \leq 10^3$, $Q = 10^3$
- 70%的数据, 任意字符串不为其他任何一个字符串的前缀;
- 100%的数据, $N \leq 40000$, $Length(S_i) \leq 10^4$, $Q \leq 10^5$
- 100%的数据, 输入字符串的总长不超过 200000。

The background features several large, stylized, colorful swirls in shades of purple, green, and blue. Interspersed among these swirls are numerous small, yellow, starburst-like shapes, some pointing towards the center and others pointing outwards, creating a dynamic and celebratory feel.

得分情况

得分情况

- 非集训队

- 60分:

- 陈立杰（浙江） 沈添笑（湖南）

- 集训队平均分：27.5

- 集训队

- 范浩强小神牛 100

The background features several large, stylized, overlapping swirls in shades of purple, green, and light blue. Interspersed among these swirls are numerous small, yellow, triangular shapes that resemble sun rays or decorative accents. The overall aesthetic is bright and modern.

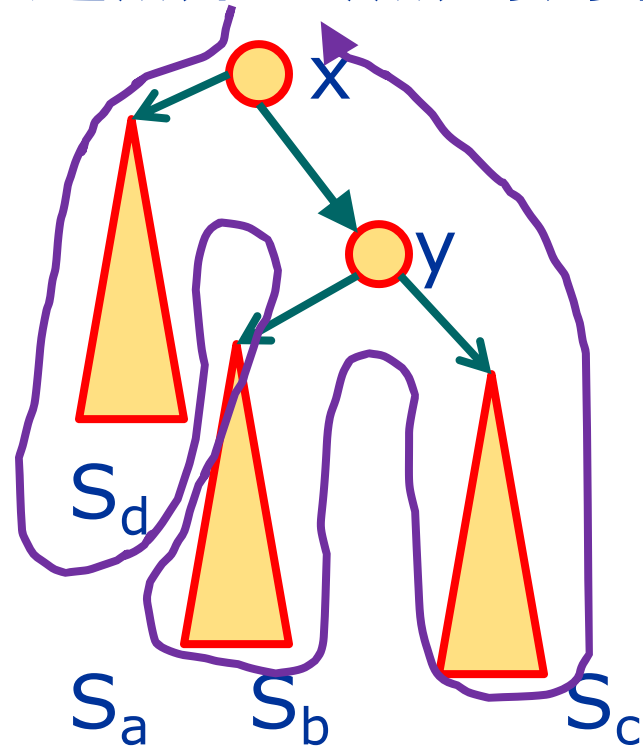
解法讨论

第一问

- 直接将所有字符串按照字典序从小到大排序即可
- 直觉上很容易发现这一点

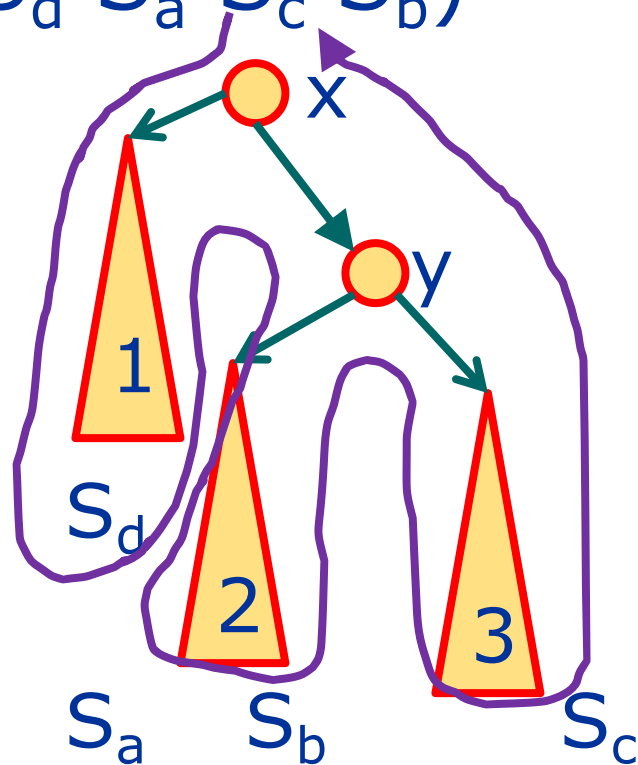
第一问

- 理性分析：
- 引入字典树，建立字典树
- 一个排列对应树上一条起点和终点均为根的路径
- 例子：
- $S_d S_a S_b S_c$
- 路径如图



第一问

- $LCP(S_a, S_b) = LCA(S_a, S_b)$ 的深度
- 容易发现重复进入子树是没有意义的
- 例: $W(S_d S_a S_b S_c) > W(S_d S_a S_c S_b)$
- 理由:
 - 对于子树2内的点, 相互间的
 - LCA必为y的孩子, 而与3之
 - 间的LCA必为y——深度浅



第一问

- 更深入的结论：
- 构建字典树之后，对树任意的DFS遍历路径所对应的字符串序列，均满足 W 最大
- 特殊的：
- 如果按照字典序遍历，则就是对字符串在线性时间内的排序

第二问

- 考虑任务：
- 由于任务的奖励是按照幂次递增的
- 策略：
- 按照任务的倒序
- 能满足就立刻满足并修正当前序列
- 不能满足就跳过

第二问

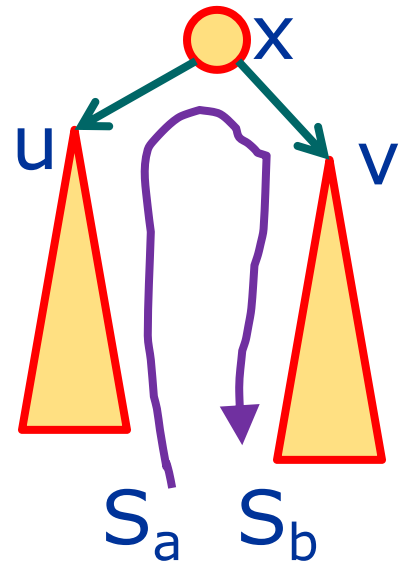
- 对于任务： S_a 要在 S_b 之前
- 对应字典树上的路径：如图
- x 为 $LCA(S_a, S_b)$

得到的限制：**第一类限制**

对称 {
- S_a 为 u 子树中最后一个访问的串
- S_b 为 v 子树中第一个访问的串

- 对于 x ，DFS时先访问 u 子树，

再访问 v 子树 **第二类限制**



第二问

- 总共两类限制：
 - 从 S_a 出发的一条向上的路径上所有的点都要求DFS时第一个走向 S_a 所在的子树
 - 对于 x 的两个孩子 u 和 v ， u 子树在 v 子树之前遍历

第二问：第一类限制

- 增加限制：
- 对于串 S_a
- 注意到每次都是对于从 S_a 出发向上的某条路径添加限制
- $\text{top}[S_a]$ 记录 S_a 的限制所达到的最高高度
- 每次覆盖从 $\text{top}[S_a]$ 向上覆盖即可
- 每个点只会至多被覆盖一次

第二问：第一类限制

- 给定任务 S_a , S_b
- 是否可以添加这对限制?
- 只要不产生矛盾即可添加
- 如何判断?
- 求解LCA为 x , 并判断 S_a 到 x 和 S_b 到 x 的路径上是否产生矛盾
- 复杂度 $O(\text{MaxLength})$

第二问：第一类限制

- 快速判断：
- 对于点 u ，如果其第一个应该访问 S_a
- S_a 所在子树为 x ，则
 - 除 x 外所有子树内的点，覆盖首先访问限制的最高高度不能达到 u （即限制 top 数组的大小）
 - x 孩子的先后关系中从 u 出发的路径为 $u \rightarrow \dots \rightarrow v$ ，若此路径不包括全部 x 的孩子，则 v 子树内的点，覆盖最后访问限制的最高高度不能达到 v
 - 路径中央的孩子 c 子树内的人一点，首先和最后访问的高度都不能到达 c

第二问：第一类限制

- 对于字典树的任意DFS序
- 子树对应DFS序中连续一段——区间覆盖
- 利用线段树套平衡树进行维护
- 维护操作：复杂度 $O(\log N)$
 - $[L,R]$ 不能超过 K
- 快速判断方法：
- 查询区间上最高高度的限制 $O(\log N)$

第二问：第二类限制

- 第二类限制：
- 对于 x 的两个孩子 u 和 v ， u 子树在 v 子树之前遍历
- 对于每个点，必然按照某种顺序依次访问其孩子的子树
- 所有孩子必然组成一条链
- 对应存在唯一的链首和链尾

第二问：第二类限制

- 我们只需维护和查询链的信息即可
- 由于每个点至多26个孩子
- 复杂度 $O(1)$
- 更进一步：
- 可以仅作不超过4次赋值和判断来完成查询和修改操作

第二问

- 维护所需数据结构：
 - 线段树
 - 链表
- 时间复杂度 $O(S+Q\log N)$
- S 为所有字符串长度之和



第二问：输出方案

- 在完成之前的顺序维护操作基础之上
- 对于每个点，DFS时按照维护完毕的链表的顺序进行遍历即可

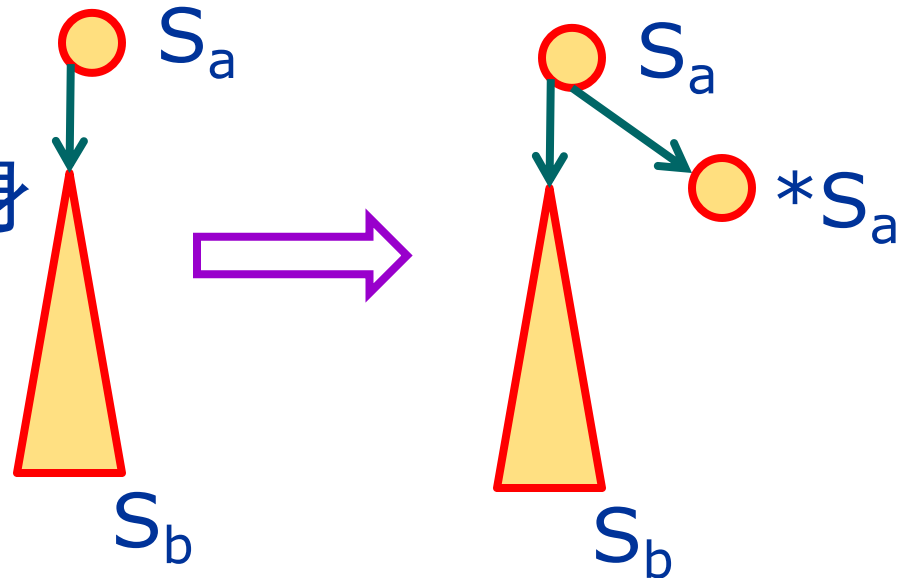
复杂度分析

- 建树 $O(S)$
- 求解LCA $O(Q\log N)$
- 第一问 $O(S)$
- 第二问 $O(S+Q\log N)$
- 第三问 $O(S)$

- 总时间复杂度 $O(S+Q\log N)$
- 空间复杂度 $O(S\log N)$

技巧

- 对于70%的数据任何串不是另一个串的前缀
- 即任何一个串在字典树中的节点均为叶子
- 如何处理 S_a 是 S_b 的前缀?
- 添加哨兵节点
- 哨兵节点对应自身



利用本题特殊性的技巧

- 利用字典树的压缩性质
- 较长字符串所对应的路径上分支很少
- 没有分支的结点此题不用处理
- 记录分支位置，直接对有分支的点进行处理

得分梯度

- 10%数据：
 - 暴力枚举排列
- 20%数据：
 - 任务唯一，一定可满足，直接输出
- 50%数据：
 - 暴力维护限制即可
 - 复杂度 $O(QMaxLength)$ 即可通过
- 70%数据：
 - 不需要处理对应点非叶子的情况

The background features several large, stylized swirls in shades of purple, green, and light blue. Interspersed among these swirls are several yellow starburst or sunburst shapes, each composed of multiple triangular rays pointing outwards. The overall aesthetic is bright and celebratory.

谢谢